

## アプリケーション開発環境 MADE における UI(User Interface) 層の設計概念

星野 准一 田淵 仁浩 山本 貴義 村岡 洋一

早稲田大学 理工学部

### 1. はじめに

MADE(MeSOD's Application Developer's Environment) は、データモデル MeSOD に基づく電子化図書 (Hyperbook) の様な高度なアプリケーションの作成を容易にするアプリケーション開発環境である。[1]

MADE は、UI(User Interface) 層、APP(Application Interface) 層 [2]、MeSOD 層 [3] の 3 層構造を有している。本稿では、UI(User Interface) 層の役割と設計概念について述べる。

### 2. UI 層に関する要件

UI 層は、Hyperbook のユーザーインターフェースを構築するためのツール群である。Hyperbook における類似検索を分かりやすく行うためには、画面上の実体に「直接的に」問い合わせができるようにするとともに、

1) 距離の取り扱い

2) 検索過程の観点

についての補助が必要となる。

1) は、距離値により表現される類似度や関連度を、分かりやすく表現することである。例えば、各県の鳥の分布の違いを見たい場合は、距離値をグラフ用紙のようなものに表示するよりも、地図の上に量的に表示してくれたほうが親切である。このように、距離の種類に応じた理解しやすい表現方法を構成することが必要になる。

2) は、検索の過程における、利用者の観点の混乱を防ぐことである。例えば利用者は、鳥を探すときに、名前、大きさなど、いろいろな観点から見てどれが近いかを相互に参照したり、ある科の鳥だけを表示して見比べたりするなどを、容易に行えなくてはならない。しかし、検索の過程で表示された様々な結果は、画面の上のスペースが限られているために、散らばったり、埋もれてしまう。そのために、少し前の結果に戻ったり、互いに参照しようとしても、どの結果がどの観点から見た結果なのか容易に理解することが出来ない。

以上のように利用者の負担となる、観点の混乱を防ぐため、検索過程における利用者の観点の切り替えについて補助する必要がある。

### 3. 解決法

次に、上記の要件を満たすユーザーインターフェースを構築する為の基本的な概念である、視覚オブジェクト、Category Plane、レイアウトオブジェクトと、その応用である Scope と Filter について説明する。視覚オブジェクトは直接的な問い合わせを実現し、レイアウトオブジェクトは距離を扱う。また Scope と Filter は検索過程における利用者の観点の補助をする。

### 3.1 基本的な操作イメージ

図 1 は、Hyperbook(鳥類図鑑)のある検索過程の断片を表している。これを使い、基本的な操作イメージについて説明する。

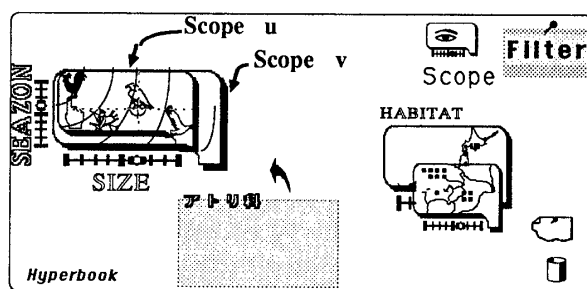


図 1 基本的な操作イメージ

図 1 の "Scope" は、「似たものが近くに見える」望遠鏡のようなものである。望遠鏡では、レンズを変えたり、高さ、方位、倍率を調節することにより目的のものを探せるが、Scope の操作イメージもそれに類似している。例えば、Hyperbook では、あたかも望遠鏡のレンズを取り替えるかのように、観点を変えられる。実際、調節する倍率や方位は、鳥の大きさや、季節に対応付けられている。

また、利用者は、"Filter" をかけることにより、ある科や目だけを選択的に見ることが出来る。しかも、望遠鏡でみたときの像は手に取ってみるようなことが出来ないが、画面上の鳥は "視覚オブジェクト" であり直接的に操作も受け付ける。

### 3.2 基本概念

#### 3.2.1 視覚オブジェクト

視覚オブジェクト (Visual Object) は、表示すべきデータと、表示機能および、イベント管理機構を備えたプログラミングオブジェクトである。これは、UI 層を構成する基本的な単位であり、画面への出力は全てこれを通して行われる。画面に表示された視覚オブジェクトは、その出力に利用者からの入力を直接的に受け付けることができる。例えば図 1 では、一つの視覚オブジェクトを通して、一匹の鳥が表示され、マウスを使った移動や、メニューの選択によるデータの問い合わせが可能である。

視覚オブジェクトには、いろいろな表し方 (名前、写真、アイコン、データの種類 (数値、文字、など) を設定できる。これを表示タイプ (Display type) と言い、場合に応じて切り替えることが容易にできる。

#### 3.2.2 Category Plane

Category Plane は、視覚オブジェクトの集合を動的に管理する「仮想的な平面」で、実画面とは独立に設定できる。各 Category Plane は、常にどの視覚オブジェクトが平面上にあるか、それ以前と比べてどんな変化があったか、を情報として持っている。

Category Plane 自身も、一つの視覚オブジェクトとして、他の Category Plane で管理することができ、より複雑な視覚オブジェクトの取り扱いを可能にする。そして、各 Category Plane に対しては、Category Plane 上の視覚オブジェクトの表示タイプなどの設定を一括して変えたり、Category Plane 間の intersection, union 等の集合演算を行える。

### 3.2.3 レイアウトオブジェクト

レイアウトオブジェクトは、Category Plane 上の視覚オブジェクト間の関係を、実体間の距離 $\{d_n\}$ を参照して動的に表現する機能を持った視覚オブジェクトである。

これは、視覚オブジェクト間の関係を、視覚オブジェクトの位置(x,y)のみでなく、心理的な要因を充分に考慮して、大きさ、形といった様々な視覚的な特徴(視覚的次元: Visual Dimension[4])に変換する。

UI 層で扱う視覚的な次元の例として、量(フォントの大きさ、分布の量など)、位置(地図などの背景と組にして提供する)、時間(カードを表示していく順番など)などが挙げられる。これらは APP 層から属性値を組み合わせることで設定できる。

## 3.3 基本概念の応用

図2に Scope と Filter の概念図を示す。Scope は、視覚オブジェクトがレイアウトされた仮想平面(Category Plane)を実画面上にマッピングするための機構である。Filter は、ある条件のものだけが通して見える、ふるいのようなものである。利用者は、実画面上で Scope に Filter を重ねることにより、より特殊な見方ができる Scope を構成できる。

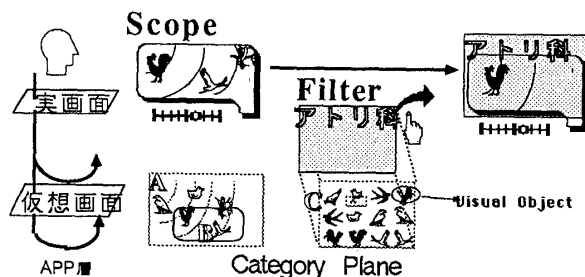


図2 Scope と Filter の概念図

### 3.3.1 Scope

Scope は、Category Plane の一部を実画面上に表示する機能を持った視覚オブジェクトである。この Scope は、Category Plane 毎に生成されるので、利用者にとって、Scope は Category Plane そのものように見える。

図2で、利用者は実画面上の Scope を通して Category Plane A の検索結果を見ている。Category Plane B は、Scope が持つ仮想平面で、Category Plane A 内の可視領域にある視覚オブジェクトを常にコピーして持っている。利用者が実画面上で Scope の位置を変えると、それに連れて可視領域も変わる。また利用者は、この Scope を通して Scope 内の視覚オブジェクトに対する直接操作を行うことができる。

利用者は、Scope を使い検索をより容易に行うことができるが、その例として、次の「Scope の重ね合わせ」について説明する。

例えば、図1で大きさの意味で近いものが見える Scope u に、季節の意味で近いものが見える Scope v を重ねたとする。利用者にとって、Scope u、Scope v を通して見るものは、両方の意味で近いものである。Scope の重なり具合は、維持管理されているので、利用者は実画面上で Scope v を取り去るだけで、いつでも Scope u の観点に戻ることができる。

実画面上で Scope が重ねられたとき、内部では Scope u の可視領域の視覚オブジェクトが Scope v の Category Plane にコピーされる。Scope v からは Scope v の Category Plane が見えていたが、利用者には実画面上の2つの Scope しか見えないので、あたかも Scope u を通して Scope v を覗いているように見える。

アプリケーションは、これらの基本操作を利用して、より高度な機能(直積、グループ化)を実装することができる。

### 3.3.2 Filter

Filter は、Scope 内の視覚オブジェクトのうち、ある条件のものだけを通す、ふるいのようなものである。Filter を重ねた Scope は、重ねる前と同様に実画面上を動かすことができる。例えば、Scope からみている検索結果に重ねると、アトリ科の鳥だけが見える Filter を構成したいとする(図2)。内部にある Category Plane C はアトリ科の鳥をデータベースに問い合わせを持っていて、初期設定で重ねたときに可視領域を表す Category Plane B と intersection を取るように設定されている。

Scope の持つ Category Plane B 上の視覚オブジェクトは、位置などの視覚オブジェクトの状態を保存しているので、利用者には Filter を重ねることによりアトリ科の鳥だけが残ったように見える。また、検索結果は Category Plane A に残っているので、Filter を外すと元の結果に戻ることができる。

## 4 おわりに

本稿では、MADE の UI (User Interface) 層とその設計概念について述べた。この層で実現した機能により、利用者は、MeSOD の提供する概念上の高度な検索を、容易にかつ分かりやすく行えるユーザーインターフェースを構築できる。

### [参考文献]

- [1] M. Tabuchi Y. Muraoka: "MeSOD: the Metric Spacial Object Data model for a multimedia application: Hyperbook", 1989, IEEE Comcon89 spring.
- [2] 山本, 田淵, 星野, 村岡: "アプリケーション開発環境 MADE とアプリケーションインターフェース層について", 情処 38 回全大.
- [3] 田淵, 村岡: "距離空間データモデル MeSOD に基づいたオブジェクト指向管理システム的设计", 情処研報 Vol.88, No.33, 88-DBS-65, 1988.
- [4] 渡辺: "大きさ, 形, 色などの間の認知的関係", 数理科学, 1988/3, pp.15-20.